Remarks

Reconsideration of the above referenced application in view of the enclosed amendment and remarks is requested. Claims 11 and 24-31 have been amended. Claim 38-42 have been added to recite further embodiments the disclosed invention. Claims 1 to 42 are now pending in the application.

ARGUMENT

Claims 24-26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicant regards has his invention. Specifically, the Examiner asserts that Claims 24-26 comprise method steps, but are written as apparatus claims. This rejection is respectfully traversed and Claims 24-26 are believed allowable based on the above amendments and the following discussion.

Claims 1-37 are rejected under 35 U.S.C. 102(e) as being anticipated by DeKoning et al. (U.S. 6,412,045) (hereinafter, "DeKoning"). This rejection is respectfully traversed and Claims 1-37 are believed allowable as amended based on the foregoing and following discussion.

Generally, DeKoning et al. teach an apparatus and method which "enables a host computer to adjust the caching strategy used for writing its write request data to storage media during execution of various software applications. The method includes the step of generating a caching-flushing parameter in the host computer. The cache flushing parameter is then transferred from the host computer to a controller which has a cache memory. Thereafter, a quantity of write request data is written from the cache memory to a storage medium in accordance with the cache-flushing parameter." (Abstract) DeKoning et al. teach caching of data to a controller having a *cache memory*. DeKoning et al. do not teach intercepting write requests and buffering them in *physical memory*, as described in Applicant's specification. Nor do DeKoning et al. teach or suggest a selective buffering mechanism to minimize activation of a memory device or storage medium to maximize battery life. Further, DeKoning et al. do not teach a technique for reading portions of data from the storage media into physical memory. In contrast, Applicant's invention enables portions of data, and supersets of the portion of data to be

read into physical memory in advance of being required in order to minimize later accesses to storage media. This allows the storage media to be powered down for a longer period, and reduces accesses to the storage media, thereby minimizing power consumption during spin up, spin down and access to the media.

As for Claim 27, the Examiner asserts that DeKoning et al. teach determining a power state of a device and selectively buffering a file system write request relating to the device based on the determined power state. The Examiner has misinterpreted the teaching of DeKoning et al. with respect to power state information. DeKoning et al. generally teach a method for using selectable caching strategies to "optimize the performance of the host computer." (See Col. 8, line 23) In contrast, Applicant's invention recites a system and method to maximize battery life. DeKoning et al. do not teach *determining the power state of a device*, as required in Claim 27 and its progeny and corresponding method claims. Applicant's claim clearly requires a *determining of power state of the device*, not whether a battery is *operational*. This is an important distinction. DeKoning et al. teach determining whether cache and alternate cache batteries are operational (BOK and ABOK flags). DeKoning et al. teach determining whether batteries are operational to avoid data loss upon power interruption: "t[T]he cache memory 39 is partitioned into at least two separate areas, a primary cache memory area 40 and an alternate cache memory area 42. The cache batteries 35, 41 exclusively power the cache memories 33, 39, respectively, in the event of a power failure or interruption to prevent the loss of data stored in the cache memories 33, 39." (Col. 3, line 67 to Col. 4, lines 1-6). DeKoning et al. teach determining whether cache batteries are operational, i.e., whether caching may take place without fear of data loss, but do not teach determining whether the host computer is operating under AC or DC (battery) power. Further, the caching schemes of DeKoning et al. do not teach maximizing battery life, but teach maximizing performance of the host computer. Thus, applying the teachings of DeKoning et al. will not result in Applicant's invention.

As for Claims 28 and 33, the Examiner asserts that DeKoning et al. teach determining whether the device is operating under battery power. This is not the case. DeKoning et al. merely teach whether an operational battery is present. DeKoning et al. do not teach whether a non-volatile storage device is actually operating under battery power. Certainly, it may be understood that the mere addition of a battery to a device that can also operate under AC power is

not determinative of whether AC or DC power is being utilized at a given time. DeKoning et al. are concerned with whether data will be lost due to a power failure which can occur if no operational battery is connected. Further, DeKoning et al. do not teach determining whether a non-volatile memory is operating under battery power, which is determinative of selectively buffering file system write request to maximize the battery life.

As for Claim 29, the Examiner asserts that DeKoning et al. teach writing one or more buffered write operations upon the occurrence of a predetermined condition. DeKoning et al. teach a cache flushing indicator which is merely a flag (on or off) to indicate whether a cache should be flushed, or written to disk. As discussed above, Claim 29 is allowable because it is dependent on Claim 27 which requires elements not shown or taught by DeKoning et al. Further, the flush indicator of DeKoning et al. is not a *detected* predetermined condition; it is merely a yes/no indicator. Newly added Claim 38 recites a further embodiment enumerating specific detected conditions which are not taught or suggested by DeKoning et al.

As for Claims 30 and 37, the Examiner asserts that DeKoning et al. teach causing a machine to deactivate the device after writing one or more buffered operations. DeKoning et al. do not teach deactivating a device, but teach completely shutting down the machine: "a[A]t a system shutdown and subsequent power down, all dirty data is quickly written to storage media..." (Col. 8, lines 48-51). Applicant's invention, on the other hand, enables the non-volatile storage devices to be individually powered down or put into a sleep state, but there is no requirement, or preference, for shutting down the entire system. Thus, applying the teaching of DeKoning et al. to Applicant's invention would require a complete shutdown after writing all of the data, rather than allowing the disks to spin down after writing a buffered write operation to the disk. Applicant's claimed invention minimizes spin up and spin down of, for instance, disks in the system. DeKoning et al. merely teach writing all buffered data to a drive before powering the system down.

Further, applying DeKoning et al. to the invention as recited in Claim 37 would cause a total loss of data. Claim 37 requires reading data from the non-volatile storage into a buffer and then deactivating the non-volatile storage. The buffered data is to be used by an application. If the system is shutdown or powered down, as taught by DeKoning et al. then this buffered data will be lost, and an application will be unable to run, as the system is powered down. This

embodiment of the invention allows a process to run on the processor while the non-volatile storage is deactivated, and thereby preserving battery life.

As for Claim 31, the Examiner asserts that DeKoning et al. teach determining a power state of a device, and selectively reading a superset of a requested file portion into physical memory. DeKoning et al. do not teach reading a superset of the requested file portion *wherein the superset of the requested file portion is logically related to the requested portion.* DeKoning et al. teach flushing cache based on a number of flags that are set to select flushing algorithms. DeKoning et al. do not teach reading a superset of the requested file portion where the superset is logically related to the requested portion.

As for Claim 32, the Examiner asserts that DeKoning et al. teach reading the entire file into physical memory. The Examiner fails to show this limitation in the cited reference. Col. 8, lines 34-67 do not teach, suggest or even hint at reading an *entire* file into physical memory, as recited by the claim. In Col. 8, lines 34-51, DeKoning et al. merely discuss immediate flushing of data, which is to write whatever is in the buffer onto a storage media. In addition, at no time is it suggested that the write buffer is to contain an entire file. In Col. 8, lines 52-67 DeKoning et al. discusses writing the cache to storage media only when there is demand for new write request data. At no time is it suggested that the write buffer is to contain an entire file, nor is it taught to *read* this file into physical memory.

As for Claim 34, the Examiner fails to show the recited limitations in the cited reference. DeKoning et al. (Col. 8, lines 52-67) do not show reading a portion of the file into a plurality of read requests. At most, DeKoning et al. teach a concatenation of *write* requests, which flush cache to the storage media. DeKoning et al. do not teach collectively causing the superset to be read *from* the device into physical memory. Further, DeKoning et al. do no teach a superset as defined in the Specification and recited in the claims.

As for Claim 35, the Examiner asserts that DeKoning et al. teach *wherein the requested file portion is read from the device and returned to a requesting process before a remainder of the superset is read into physical memory.* In fact, DeKoning et al. teach (at Col. 10, lines 37-48 and 65-67) writing a quantity of write request data before writing another quantity of write request data, but writing only after determining that the amount of data in the write request cache exceeds some threshold. The cited reference does not teach *reading* the data from the device, but

only writing the data to the device. Further, the cited reference does not teach a superset of a requested file, *wherein the superset of the requested file portion is logically related to the requested portion.* Data written from the cache to a storage device (flushing) is not described as related data. It is merely an unrelated collection of write requests. Moreover, flushing a cache in this fashion will not result in prolonging battery life as recited by Applicant's invention.

As for Claim 36, the cited reference does not anticipate Applicant's claimed invention. The Examiner asserts that DeKoning et al. teach *accessing the superset read into physical memory to fulfill a subsequent file system request to read a portion of the file.* DeKoning et al. (Col. 6, lines 23-25) teach write-back caching in the case of a 2-minute warning form a UPS. Once again, DeKoning et al. teach a method for flushing the cache (writing) the portion to a storage device. DeKoning et al. do not teach *reading* portions of data, in an order or amount, into physical memory to fulfill read requests.

As for Claims 1-23 (corresponding method claims) and Claims 24-26 (corresponding system claims), they are believed allowable based on the amendments and foregoing discussion. Further, blanket rejection of Claims 1-26 is improper as they do not correspond exactly to rejected claims 27-37. Therefore, because no explicit grounds of rejection have been supplied, Applicant respectfully requests that if the Examiner should maintain the rejections to Claims 1-23 and Claims 24-26, that a new non-final Office Action be provided.

Specifically, regarding Claim 1 and its progeny, DeKoning et al. do not teach determining whether the device is activated or inactivated. This determination is important to minimize disk spin up or premature spin down. The cited references in DeKoning et al. teach only whether cache memory has an operational battery, but not whether the storage media is running, sleeping, on or off.

Regarding Claim 9, at no time do cited references in DeKoning et al. teach that a requested write operation corresponds to an entity registered to participate in the method of controlling device write operations. Nor does the Examiner show this limitation anywhere in the cited reference, or address this limitation in any way. Applicant's claimed invention enables an Application to take advantage of read/write policies for selective buffering. DeKoning et al. do not teach or suggest this limitation.

Claim 10 and newly added Claim 41 require that buffering the write operation may delete a previously buffered write request if the new request is to the same memory location. DeKoning et al. do not teach the deletion of write requests from the buffer prior to writing the buffer to storage media. Nor does the Examiner show this limitation anywhere in the cited reference, or address this limitation in any way.

Regarding Claim 11 and its progeny, Applicant's invention may detect a limited power condition, for instance, low battery, and read more data from storage than was requested to avoid future reads from the storage device. This reading of additional *related* data makes a larger portion of a requested file available to an application without additional read accesses forcing the storage media to become activated (i.e., spin up and use more battery) unnecessarily, as the additional data is requested by the application. Instead, the application retrieves the data from physical memory. DeKoning et al. does not teach efficient reading of data from a storage medium, but instead teach flushing of cache memory to the storage medium.

Thus, all claims remaining in the application are now allowable.

CONCLUSION

In view of the foregoing, Claims 1 to 42 are all in condition for allowance. If the Examiner has any questions, the Examiner is invited to contact the undersigned at (703) 633-6845. Early issuance of Notice of Allowance is respectfully requested. Please charge any shortage of fees in connection with the filing of this paper, including extension of time fees, to Deposit Account 50-0221 and please credit any excess fees to such account.

Respectfully submitted,

Dated: 10/25/2004

Joni D. Stutman-Horn
Patent Attorney
Intel Corporation
Registration No. 42,173
(703) 633-6845

c/o Intel Americas, Inc.
4030 Lafayette Center Drive
MS CY-LF3
Chantilly, VA 20151